Alternative KPSO-Clustering Algorithm

Fun Ye* and Ching-Yi Chen

Department of Electrical Engineering, Tamkang University, Tamsui, Taiwan 251, R.O.C.

Abstract

This paper presents an evolutionary particle swarm optimization (PSO) learning-based method to optimally cluster N data points into K clusters. The hybrid PSO and K-means algorithm with a novel alternative metric, called Alternative KPSO-clustering (AKPSO), is developed to automatically detect the cluster centers of geometrical structure data sets. The alternative metric is known has more robust ability than the common-used Euclidean norm. In AKPSO algorithm, the special alternative metric is considered to improve the traditional K-means clustering algorithm to deal with various structure data sets. For testing the performance of the proposed method, this paper will show the experience results by using several artificial and real data sets. Simulation results compared with some well-known clustering methods demonstrate the robustness and efficiency of the novel AKPSO method.

Key Words: Clustering, Particle Swarm Optimization, K-means

1. Introduction

Cluster analysis has become an important technique in exploratory data analysis, pattern recognition, machine learning, neural computing, and other engineering. The clustering aims at identifying and extracting significant groups in underlying data. In the field of clustering, K-means algorithm is the most popularly used algorithm to find a partition that minimizes mean square error (MSE) measure. Although K-means is an extensively useful clustering algorithm, it suffers from several drawbacks. The objective function of the K-means is not convex and hence it may contain local minima. Consequently, while minimizing the objective function, there is possibility of getting stuck at local minima (also at local maxima and saddle point) [1]. The performance of the K-means algorithm depends on the initial choice of the cluster centers. Besides, the Euclidean norm is sensitive to noise or outliers. Hence K-means algorithm should be affected by noise and outliers [2]. Recently, the use of global optimization techniques such as Simulated Annealing and Genetic Algorithm (GA) has emerged in the clustering fields [3]. They are capable of searching for optimal or near-optimal solutions on complex, large spaces of possible solutions. Because of this advantage, it may represent another useful tool in the field of cluster analysis.

Particle swarm optimization (PSO) is a population-based algorithm [4–6]. This algorithm simulates bird flocking or fish schooling behavior to achieve a self-evolution system. It can automatically search the optimum solution in the solution space. But the searching process isn't randomness. According to the different problems, it decides the searching way by the fitness function. Unlike other evolutionary learning algorithms, PSO needs smaller parameters to decide. PSO can be easily implemented, and has stable convergence characteristic with good computational efficiency. In traditional clustering analysis methods such as k-means and fuzzy c-means, it will get easily stuck at local minima and be not robust to cluster complex data sets. To overcome these problems, we provide an evolutionary-based clustering method by PSO. It has the higher ability to find the near-optimal solutions in the search space.

*Corresponding author. E-mail: fyee@mail.tku.edu.tw

The rest of the paper is organized as follows. We introduce the PSO algorithm in Section II. The proposed algorithm is explained in Section III. The algorithm is test on several artificial and real data sets. Details of simulations and results are presented in Section IV. We conclude with a summary of the contributions of this paper in Section V.

2. Particle Swarm Optimization

PSO is an evolutionary computation technique developed by Kenney and Eberhart in 1995 [4]. The method has been developed through a simulation of simplified social models. PSO is based on swarms such as fish schooling and bird flocking. According to the research results for bird flocking, birds are finding food by flocking (not by each individual). Like GA [3,7], PSO must also have a fitness evaluation function that takes the particle's position and assigns to it a fitness value. The position with the highest fitness value in the entire run is called the global best (*gbest*). Each particle also keeps track of its highest fitness value. The location of this value is called its personal best (*pbest*). The basic algorithm involves casting a population of particles over the search space and remembering the best (most fit) solution encountered. At each iteration, every particle adjusts its velocity vector, based on its momentum and the influence of both its best solution and the best solution of its neighbors, then computes a new point to examine. The studies shows that the PSO has more chance to "fly" into the better solution areas more quickly, so it can discover reasonable quality solution much faster than other evolutionary algorithms. The original PSO formulate is described as [4–6]:

$$V_{i,d}(t+1) = \tau \cdot V_{i,d}(t) + c_1 * rand(\ ) * (pbest_{i,d}(t) - X_{i,d}(t)) + c_2 * rand(\ ) * (gbest_d(t) - X_{i,d}(t)) \quad (1)$$

$$X_{i,d}(t+1) = X_{i,d}(t) + V_{i,d}(t+1) \quad (2)$$

where d is the number of dimensions (variables), i is a particle in the population, *gbest* is the best position vector found in a certain neighborhood of the particle, V is the velocity vector, X is the position vector, τ is the inertia factor, and *pbest* is the position vector for a particle's best fitness yet encountered. Parameters c_1 and c_2 are the cognitive and social learning rates, respectively. These two rates control the relative influence of the memory of the neighborhood and the memory of the particle.

3. Clustering with Pso Algorithm

3.1 Clustering Analysis

Clustering analysis is a technology which can classify the similar sample points into the same group from a data set [8]. It is a branch from multi-variable analysis and unsupervised learning rule in the pattern recognition. For space S which has the K groups and the N points $\{x_1, x_2, \ldots, x_N\}$, the definition of the clustering analysis is as follow:

1) Data set $X = \{x_1, x_2, \ldots, x_N\}$, the *ith* data point x_i is a vector in n-dimensional space, the number of the data points $N < \infty$.
2) Cluster set $C = \{C_1, C_2, \ldots, C_K\}$, K represents the cluster number by partitioning X. These K nonempty sets completely disjoint. Then

$$\begin{cases} C_i \neq \phi, \quad for \quad i = 1,2,\ldots K \\ C_i \cap C_j = \phi, i \neq j \\ \bigcup_{i=1}^{K} C_i = X, \end{cases} \quad (3)$$

where ϕ is an empty set.

The target of cluster analysis is the highest similar characteristic data in each cluster C_i and the least similar characteristic data in the other clusters. Each cluster C_i can get a n-dimensional cluster center z_i. It is the center of the whole data points in C_i. The iteration algorithm to calculate the cluster center z_i is as below [9].

Step 1) Given a cluster center set $Z_m = \{z_1, z_2, \ldots, z_K\}$, obtained from the m^{th} iteration, assign each data point to the closet cluster center.

Step 2) Obtain the new cluster center Z_{m+1} by computing the cluster center of each cluster based on partitioning of Step 1.

Note that the iteration algorithm will be terminated when the final cluster centers are created.

In many clustering techniques, the K-means (gener-

ally called Hard c-means) algorithm is one of well-known hard clustering techniques. It can allocate the data point x_i to the closest cluster center z_j by using Euclidean distances.

$$D = \|x_i - z_j\|,\ i \in \{1,2,...,N\},\ j \in \{1,2,...,K\} \tag{4}$$

A good method will cluster data set $X = \{x_1, x_2, \ldots, x_N\}$ into K well partitions with $2 \le K \le N-1$. When we have an unlabelled data set, it is very important to define a objective function for a clustering analysis method. Intuitively, each cluster shall be as compact as possible. Thus, the objective function of the K-means algorithm is created with the Euclidean norm. It represents as below:

$$J_E = \sum_{j=1}^{K} \sum_{x_i \in C_j} \|x_i - z_j\|^2 \tag{5}$$

where z_j is the jth cluster center. The necessary condition of the minimum J_E is

$$z_j = \frac{1}{N_j} \sum_{x_i \in C_j} x_i\ ,\ j = 1,2,...,K \tag{6}$$

where N_j is the number of points belonging to cluster C_j.

3.2 Alternative KPSO-clustering

We will now introduce how to utilize Alternative KPSO-clustering (AKPSO) to detect the cluster centers of a given data set. An initial population of solutions called particles is randomly generated, and each string is a sequence of real numbers representing the K cluster centers. For an n-dimensional space, the length of a particle is $K*n$ [10].

Figure 1 is an example of the encoding of the single particle in the PSO initial population. Let $n = 3$, $K = 3$, the string of this particle represents three cluster centers [(61, 42.3, 35.7), (75.1, –20, 15) and (9.68, 21.2, 18.7)].

After the encoding of the string of the particles, the execution of AKPSO is as follow:

Step 1) Initialize position vector X and associated velocity V of all particles in the population randomly.

Step 2) Evaluate the fitness function for each particle. We use metric function which proposes by [2] to measure the similarity /dissimilarity between the various elements of a data set. For each data point x_i, we assign point x_i, $i \in \{1,2, \ldots, N\}$ to cluster C_j, $j \in \{1,2,\ldots,K\}$ iff

$$1 - \exp(-\beta \|x_i - z_j\|^2) = \min_p \{1 - \exp(-\beta \|x_i - z_p\|^2)\},\quad p = 1,2,...,K \tag{7}$$

$$\text{where } \beta = \left(\frac{\sum_{i=1}^{N} \|x_i - \bar{x}\|^2}{N} \right)^{-1},\ \bar{x} = \frac{\sum_{i=1}^{N} x_i}{N} \tag{8}$$

In our research, the selected fitness function of the AKPSO is given by:

$$fitness = \frac{k_o}{J_o + \sum_{j=1}^{K} \sum_{x_i \in C_j} \left\{ 1 - \exp(-\beta \|x_i - z_j\|^2) \right\}} \tag{9}$$

where the k_o is a positive constant, and J_o is a small-valued constant.

Step 3) Compare every particle's fitness value with previous particle's best solution (*pbest*). If current

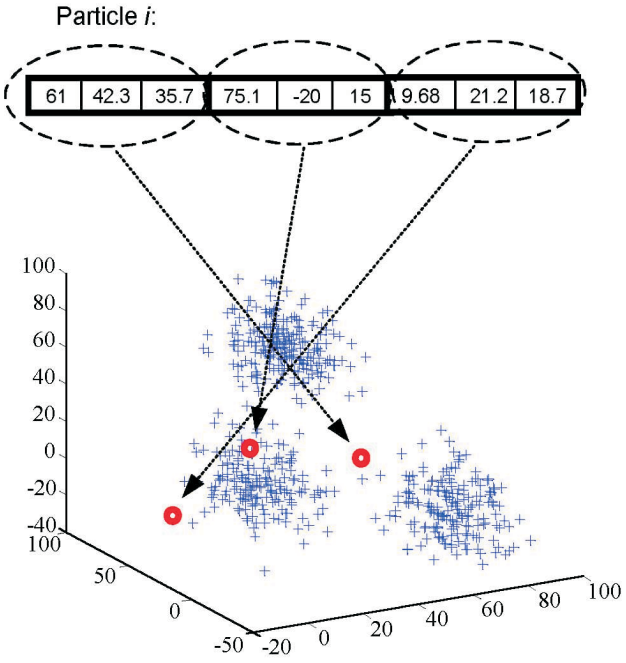

Figure 1. The encoding of the single particle in the PSO initial population.

solution is better than previous value (*pbest*), then update *pbest* with current solution.

Step 4) Compare fitness evaluation with the population's overall previous best. If current value is better than the *gbest* (the global version of the best value), then reset *gbest* to the current particle's value and position.

Step 5) Use the one step of K-means algorithm to replace the result of the *gbest*. The cluster centers encoded in the *gbest* are replaced by the mean points of the respective clusters [11]:

$$z_j^* = \frac{1}{N_j} \sum_{x_i \in C_j} x_i,\ j = 1,2,...,K \tag{10}$$

where N_j is the number of points belonging to cluster C_j. The effect of the K-means algorithm is to direct the best solution towards the area of the training data. The drawback of the hybridization is that the running time considerably grows as the number of K-means step increases. For better convergence and lower computing time purpose, the Step 5 work in the initial five iterations (or less) is enough.

Step 6) Change velocities and position with Eq. (1) and Eq. (2).

Step 7) Repeat Step2)-Step6) until the predefined number of iterations is completed.

4. Simulation Results

The effectiveness of the proposed algorithm is illustrated for clustering data sets with different geometrical structures. The parameters of the proposed AKPSO for all examples are defined as follows: c1 = c2 = 1.5, k_o = 50, J_o = 0.1 and τ = 0.75.

4.1 Experiment 1

First, we demonstrate the clustering ability on four data sets with different dimension and shape. It can show the clustering ability by using the proposed method.

Example 1. The data set in Figure 2(a) is composed by 250 two-dimensional data points [12]. It contains five spherical clusters, and the clusters are close to each other.

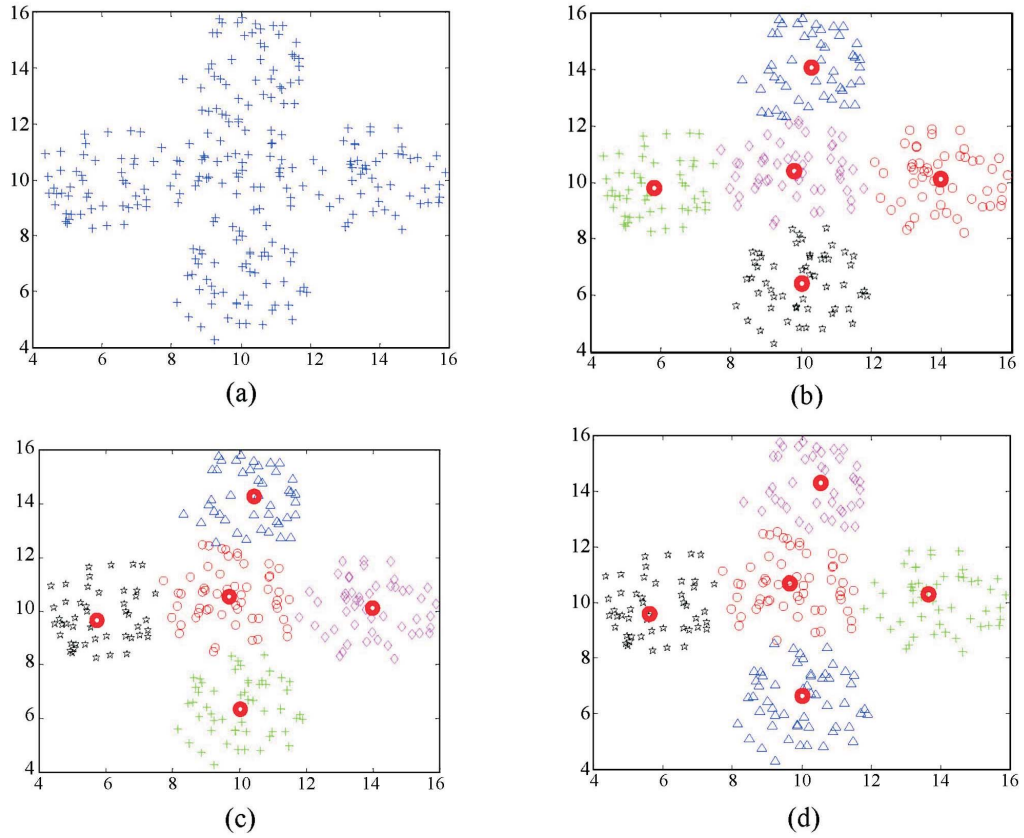

Figure 2. (a) The data set used in example 1. The clustering results achieved by the (b) K-means; (c) Fuzzy c-means; (d) AKPSO.

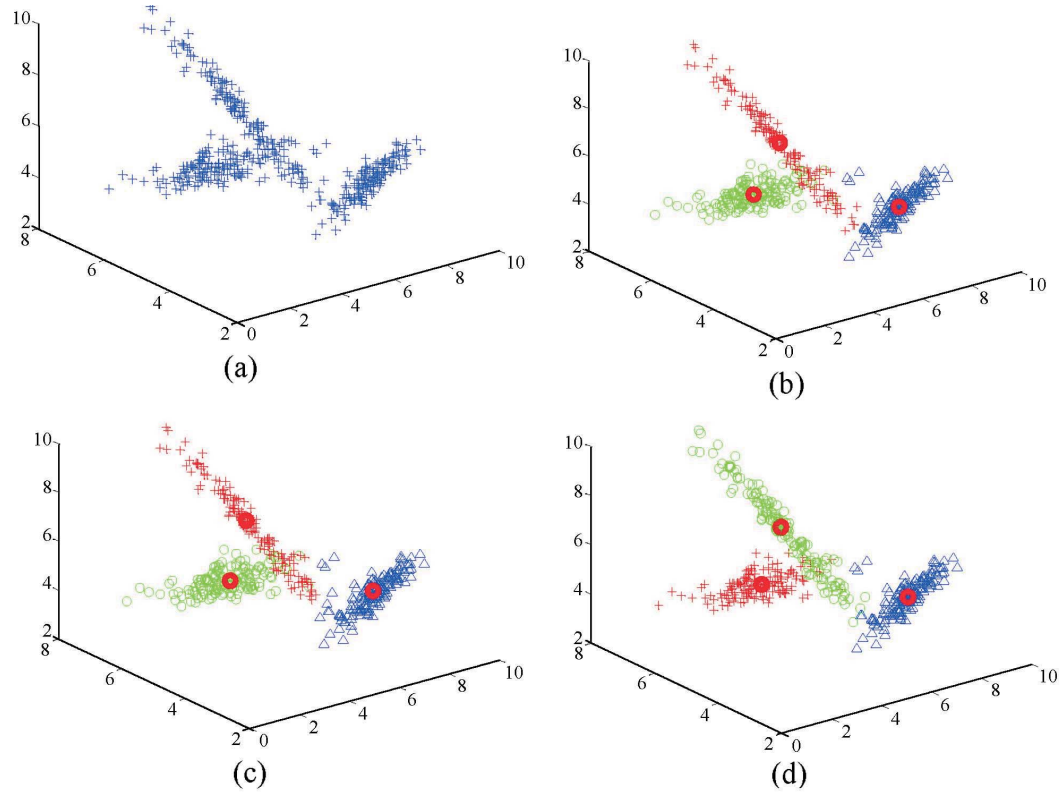

Figure 3. (a) The data set used in example 2. The clustering results achieved by the (b) K-means; (c) Fuzzy c-means; (d) AKPSO.

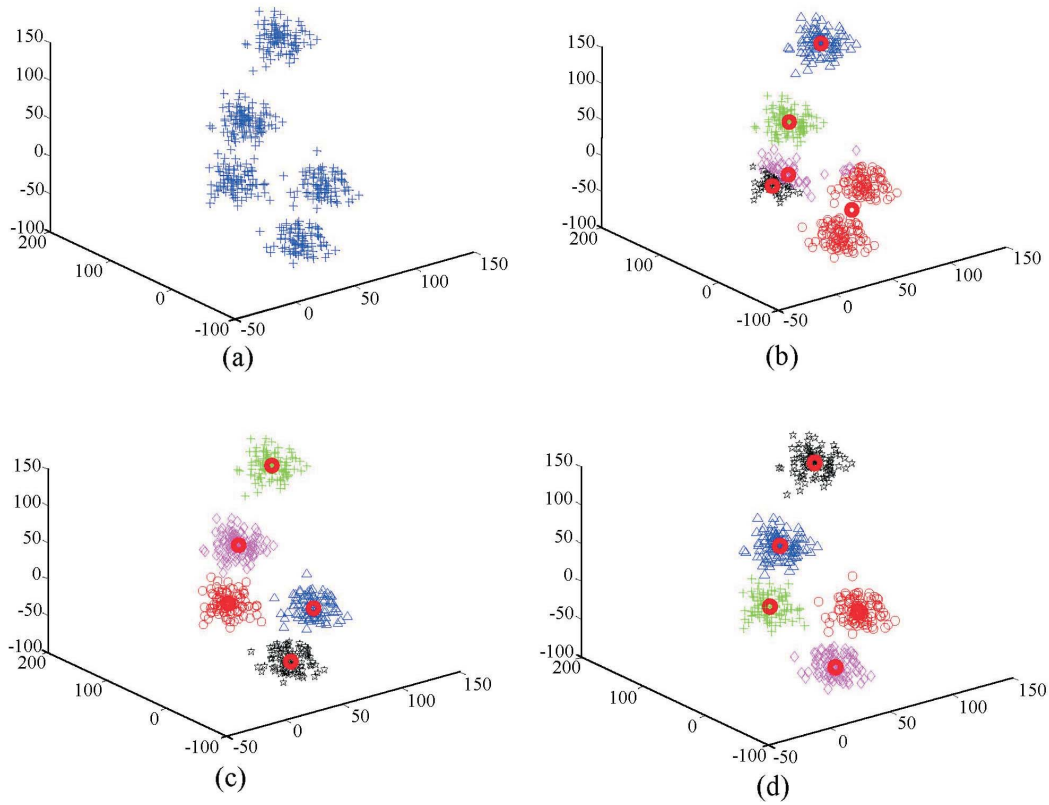

Figure 4. (a) The data set used in example 3. The clustering results achieved by the (b) K-means; (c) Fuzzy c-means; (d) AKPSO.

Figure 2(d) shows the clustering result of the AKPSO algorithm. It is similar to the results of K-means and Fuzzy c-means (Figure 2(b) and (c)).

Example 2. This data set in Figure 3(a) contains 450 data points distributed on three linear clusters, and these three clusters are overlapping in the two-dimensions of the cubic space [13]. Both of the K-means and the AKPSO have similar clustering results (Figure 3(b) and (d)), and the clustering results achieved by the K-means and the AKPSO are a little bit better than that achieved by the Fuzzy c-means algorithm (Figure 3(c)).

Example 3. A data set which contains five spherical clusters is shown in Figure 4(a). It is composed by 500 three-dimensional data points. The clustering result achieved by the K-means is shown in Figure 4(b). The clustering results achieved by the Fuzzy c-means and AKPSO are illustrated in Figure 4(c) and 4(d), respectively. Here, both the Fuzzy c-means and the AKPSO

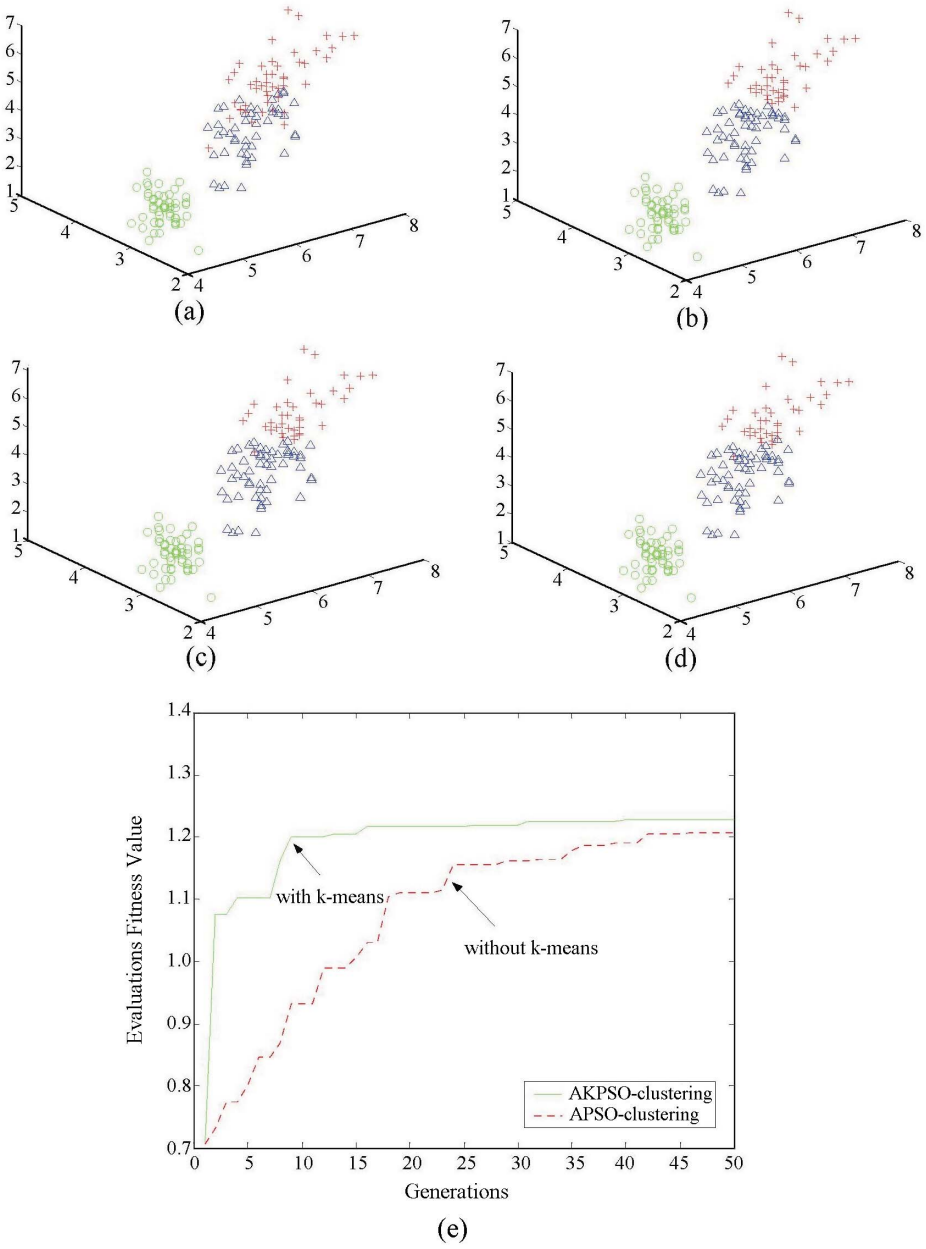

Figure 5. (a) The three-dimensional plot for the four-dimensional Iris data: iris setosa (O), iris versicolor (Δ), and iris virginica (+). The clustering results achieved by the (b) K-means; (c) Fuzzy c-means; (d) AKPSO. (e) The proposed algorithm with and without one step of K-means algorithm (by using IRIS data), and the population size P is taken to be 40.

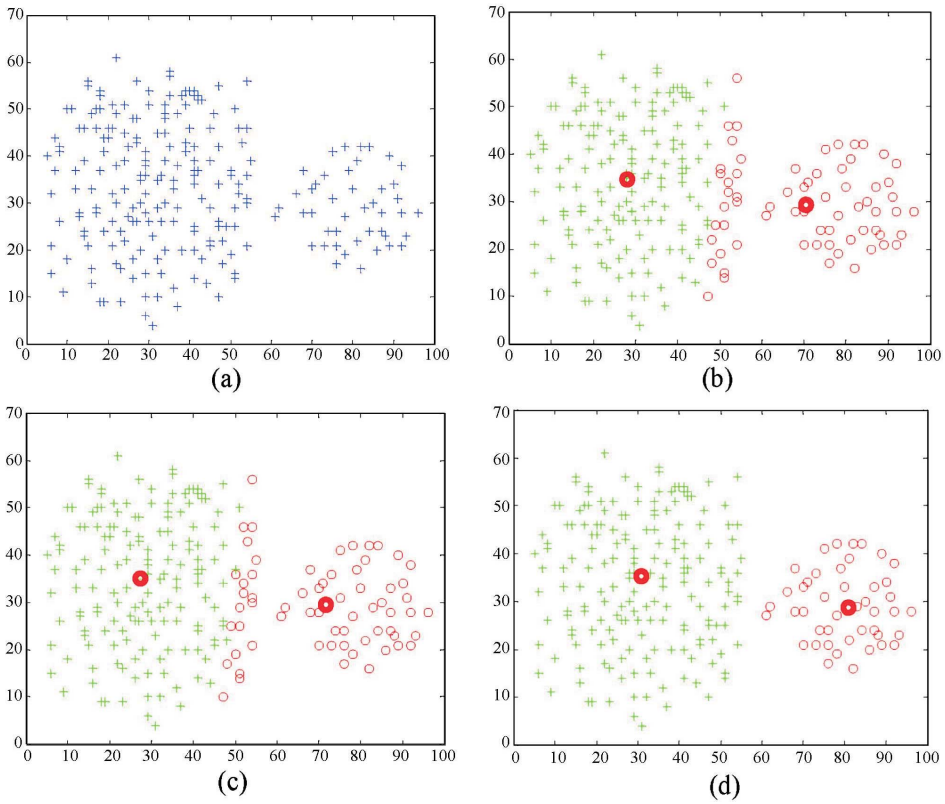

Figure 6. (a) The data set used in example 5. The clustering results achieved by the (b) K-means; (c) Fuzzy c-means; (d) AKPSO.

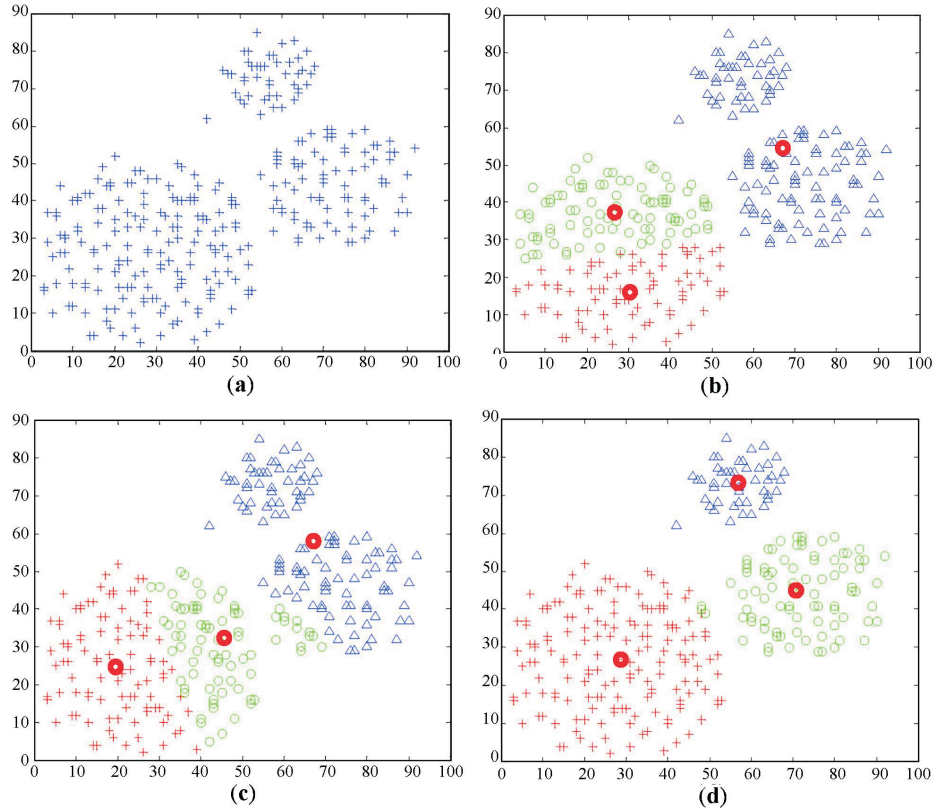

Figure 7. (a) The data set containing three spherical clusters with different sizes. The clustering results achieved by the (b) K-means; (c) fuzzy c-means; (d) AKPSO.

can correctly cluster this data set.

Example 4. The Iris data set has three subsets (i.e. Iris setosa, Iris versicolor, and Iris virginica). There are total 150 data points in the data set. Each class has 50 patterns. The three-dimensional plot for the four-dimensional Iris data set shown in Figure 5(a). The K-means, Fuzzy c-means and AKPSO clustering results are shown in Figure 5(b), 5(c) and 5(d), respectively. They have similar clustering results. Figure 5(e) illustrates the convergence characteristics of AKPSO (with K-means) and APSO (without K-means). As the results, AKPSO converges obviously faster then APSO.

4.2 Experiment 2

In this experiment, we test the clusters with different sizes to prove the ability of classification by using our proposed method.

Example 5. We add more points to the left spherical cluster (Figure 6(a)). The K-means and Fuzzy c-means results have many misclassified data (Figure 6(b) and (c)). But AKPSO can cluster these two clusters correctly (Figure 6(d)).

Example 6. The data set shown in Figure 7(a) is intuitively partitioned into three clusters. The size of the upper cluster is apparently smaller than those of the remaining two clusters. Obviously, the clustering results shown in Figure 7(d) is better than the clustering results shown in Figure 7(b) and 7(c).

4.3 Experiment 3

The goal of this experiment is to verify the reasonability by using our proposed method to detect the cluster center location.

Example 7. The artificial data sets on two planes in which one is zero-section (i.e. z = 0) plane and another one-section (i.e. z = 1) plane as shown in Figure 8(a) [2]. In Figure 8(d), AKPSO found cluster centers on the z = 0 and z = 1. In Figure 8(b) and (c), the cluster centers are not in these two data planes, there are still many misclassified data for K-means and Fuzzy c-means.

Table 1 is the performance comparison between K-means, Fuzzy c-means and AKPSO. The distortion measure defined in Eq. (5) calculated by AKPSO is better than those of other algorithms. In many times

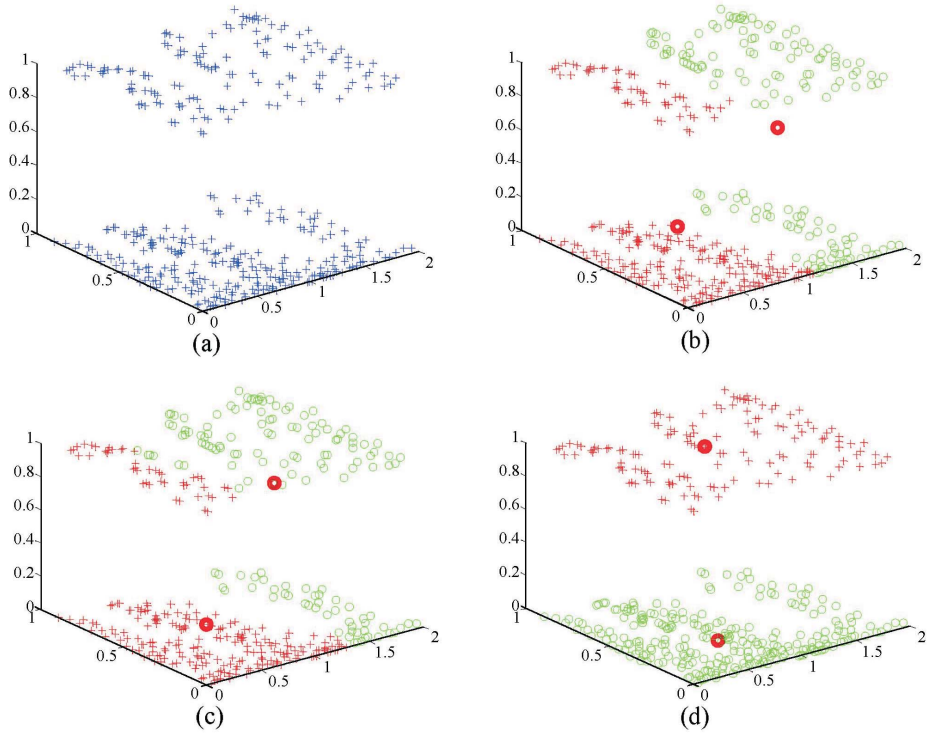

Figure 8. (a) The data set used in example 7. (b) The clustering result achieved by K-means, and the cluster centers are [(0.4838,0.3822,0.2222), (1.5070,0.4489,0.6027)]. (c) The clustering result achieved by Fuzzy c-means, and the cluster centers are [(0.5491,0.3608,0.1063), (1.3690,0.4874,0.7590)]. (d) The clustering result achieved by AKPSO, and the cluster centers are [(0.9890,0.5760,1.0000), (0.7073,0.3129,0.0000)].

Table 1. Comparison of K-means, Fuzzy c-means, and AKPSO

	K-means		Fuzzy c-means		AKPSO (Pop. size $p = 40$)	
Data set	$J_E^{1/2}$	Cluster centers	$J_E^{1/2}$	Cluster centers	$J_E^{1/2}$	Cluster centers
Example 1	328.4659	(13.9876, 10.1164)	327.5613	(13.9860, 10.1353)	326.9552	(10.5779, 14.3184)
		(5.8100, 9.7929)		(10.0232, 6.3699)		(10.0967, 6.6486)
		(10.2843, 14.0613)		(5.7049, 9.6745)		(9.7037, 10.6499)
		(9.7977, 10.4025)		(9.6913, 10.5300)		(13.6976, 10.2977)
		(10.0217, 6.4375)		(10.4229, 14.2644)		(5.5913, 9.5773)
Example 2	507.5669	(3.9864, 5.0367, 7.2676)	509.9258	(4.0050, 5.2665, 7.4035)	506.9080	(4.8219, 6.5025, 3.9561)
		(4.8219, 6.5025, 3.9561)		(6.8847, 3.5193, 4.7686)		(6.9122, 3.5056, 4.6938)
		(6.9494, 3.5165, 4.6865)		(4.8400, 6.4231, 3.9905)		(3.9861, 5.0695, 7.2952)
Example 3	11413	(50.3251, –30.7423, –30.7423)	7595.7	(29.7088, –50.9759, –50.9759)	7588.9	(29.7591, –51.3503, –51.3503)
		(41.3610, 59.8459, 59.8459)		(100.4467, 119.8950, 119.8950)		(69.5475, –11.0075, –11.0075)
		(100.4467, 119.8950, 119.8950)		(69.4938, –10.7688, –10.7688)		(41.4634, 60.0187, 60.0187)
		(17.6656, 14.4181, 14.4181)		(8.6852, 12.4977, 12.4977)		(100.7888, 120.2491, 120.2491)
		(0.7778, 8.9529, 8.9529)		(41.1584, 59.6720, 59.6720)		(8.5566, 12.6721, 12.6721)
Example 4	98.1872	(6.8068, 3.1205, 5.5227, 1.9818)	96.9280	(5.8890, 2.7612, 4.3640, 1.3973)	96.7551	(5.0127, 3.3995, 1.4722, 0.2341)
		(5.8339, 2.6768, 4.4214, 1.4357)		(6.7749, 3.0524, 5.6467, 2.0535)		(6.6982, 3.0651, 5.5957, 2.1131)
		(5.0060, 3.4180, 1.4640, 0.2440)		(5.0036, 3.4030, 1.4850, 0.2515)		(5.9559, 2.8186, 4.4424, 1.4225)
Example 5	4037.1	(28.1569, 43.7549)	3659.7	(27.2686, 35.0305)	3614.7	(80.9615, 28.3741)
		(51.0331, 24.2975)		(71.6859, 29.4891)		(30.6171, 35.4287)
Example 6	4535.7	(30.1500, 16.1500)	4501.4	(19.4702, 24.7734)	4210.5	(70.6181, 45.0851)
		(26.5116, 37.4302)		(66.8719, 58.1697)		(56.7468, 73.3289)
		(66.9764, 54.7244)		(45.4299, 32.5066)		(28.5811, 26.9885)
Example 7	228.1632	(0.4838, 0.3822, 0.2222)	225.5565	(0.5491, 0.3608, 0.1063)	221.6049	(0.9890, 0.5760, 1.0000)
		(1.5070, 0.4489, 0.6027)		(1.3690, 0.4874, 0.7590)		(0.7073, 0.3129, 0.0000)

test, AKPSO can approximate the final convergence results. It can avoid such K-means easy to getting stuck in a poor local optima.

5. Conclusion

In summary, a PSO algorithm to solve clustering problems has been developed in this paper, called Alternative KPSO-clustering (AKPSO). We developed an evolutionary-based clustering technique by hybridizing the K-means algorithm and PSO. It can be considered as a viable and an efficient heuristic to find optimal or near-optimal solutions to clustering problems of allocating N data points to K clusters. The proposed method is very efficient and simple to implement for clustering analysis when the number of clusters is known a priori. According to the simulation results of the proposed approach to a number of data sets with different geometrical properties, it is clear that the proposed approach is more robust than the traditional clustering analysis algorithms.

Acknowledgment

Authors would like to thank the National Science Council of Taiwan for their financial support on this research by the grant NSC92_2213_E_032_027.

References

[1] Selim, S. Z. and Ismail, M. A., "K-means Type Algorithms: A Generalized Convergence Theorem and Characterization of Local Optimality," *IEEE Trans. Pattern Anal. Mach. Intell.* Vol. 6, pp. 81–87 (1984).

[2] Wu, K.-L. and Yang, M.-S., "Alternative C-means Clustering Algorithms," *Pattern Recognition*, Vol. 35, pp. 2267–2278 (2002).

[3] Maulik, U. and Bandyopadhyay, S., "Genetic Algorithm-based Clustering Technique," *Pattern Recognition* Vol. 33, pp. 1455–1465 (2000).

[4] Kennedy, J. and Eberhart, R., "Particle Swarm Optimization," *Proc. of IEEE International Conference on Neural Networks (ICNN)*, Perth, Australia, Vol. 4, pp. 1942–1948 (1995).

[5] Eberhart, R. and Kennedy, J., "A New Optimizer Using Particle Swarm Theory," *Proc. 6th Int. Symposium on Micro Machine and Human Science*, pp. 39–43 (1995).

[6] Eberhart, R. C. and Shi, Y., "Particle Swarm Optimization: Developments, Applications and Resources," *Proceedings of the IEEE Congress on Evolutionary Computation (CEC 2001)*, Seoul, Korea (2001).

[7] Filho, J. L. R., Treleaven, P. C. and Alippi, C., "Genetic Algorithm Programming Environments," *IEEE Comput*. Vol. 27, pp. 28–43 (1994).

[8] Anderberg, M. R., *Cluster Analysis for Application*, Academic Press, New York, NY, U.S.A. (1973).

[9] Allen Gersho and Robert M. Gray, *Vector Quantization and Signal Compression*, Springer Verlag, The Netherlands (1992).

[10] Chen, Ching-Yi. and Ye, Fun., "Particle Swarm Optimization Algorithm and Its Application to Clustering Analysis," *IEEE ICNSC 2004*, Taipei, Taiwan, R.O.C., pp. 789–794 (2004).

[11] Chen, Ching-Yi. and Ye, Fun., "K-means Algorithm Based on Particle Swarm Optimization," *2003 International Conference on Informatics, Cybernetics, and Systems*, I-Shou University, Taiwan, R.O.C. pp. 1470–1475 (2003).

[12] Bandopadhyay, S. and Maulik, U., "Genetic Clustering for Automatic Evolution of Clusters and Application to Image Classification," *Pattern Recognition*, Vol. 35, pp. 1197–1208 (2002).

[13] Wong, C. C. and Lin, B. C., "Neighbor-based Clustering Algorithm," *International Journal of Electrical Engineering*, Vol. 11, pp. 173–181 (2004).

Manuscript Received: Jan. 28, 2005
Revision Received: Mar. 14, 2005
Accepted: Apr. 12, 2005